CLAIMS

1. A method of detecting failure in one or more file systems supported by a non-volatile memory, the method comprising:

persistently storing critical information in the non-volatile memory, the critical information indicating a status of each of the one or more file systems;

in response to a predetermined event, obtaining critical information associated with the one or more file systems stored in the non-volatile memory immediately before the predetermined event;

determining if one of the one or more file systems has crashed based on the critical information; and

determining if one of the one or more file systems is mounted improperly if it is determined that the one of the one or more file systems has crashed.

2. The method according to claim 1, wherein the critical information associated with each of the one or more file systems comprises a data structure indicating the status of a corresponding one of the one or more file systems.

3. The method according to claim 1, further comprising initiating a failure recovery process when it is determined that one of the one or more file systems is mounted improperly.

4. The method according to claim 3, wherein the critical information indicates at least one of a normal-off status, a running status and an abnormal-off status, and

wherein the critical information is configured to change from the normal-off status to the running status during a start-up procedure, and

wherein the critical information is configured to change to the normal-off status after a shutting-down procedure, and

wherein the critical information is configured to change to the abnormal-off status when a failure is detected in the file system.

5. The method according to claim 4, further comprising determining if the file system has been properly shut down after the shutting down procedure.

6. The method according to claim 1, further comprising persistently storing metadata in an area of the non-volatile memory such that the metadata will not be overwritten, wherein the metadata is used to reboot one of the one or more file systems when it is determined that the one of the one or more file systems is mounted improperly.

7. The method according to claim 1, wherein the predetermined event is one during which one of the one or more file systems has crashed and a failure is detected in the one of the one or more file-systems.

8. A method of rebooting one or more file systems supported by a non-volatile memory, the method comprising:

persistently storing critical information in the non-volatile memory, the critical information indicating a status of each of the one or more file systems;

in response to a predetermined event, obtaining critical information associated with the one or more file systems stored in the non-volatile memory immediately before the predetermined event;

determining if one of the one or more file systems has crashed based on the critical information; and

rebooting from metadata in the non-volatile memory if it is determined that the one of the one or more file systems has crashed, wherein the one of the one or more file systems is in a duplex state before the predetermined event and is configured to switch to a single state when it is determined that the one of the one or more file systems has crashed to allow the metadata to be flushed to a block device.

9. The method according to claim 8, wherein the file system is further configured to switch back to the duplex state when the metadata is flushed to the block device.

10. A non-transitory computer readable storage medium including a set of instructions executable by a processor, the non-transitory computer readable storage medium for detecting failure in one or more file systems supported by a non-volatile memory, the set of instructions operable to:

persistently store critical information in the non-volatile memory, the critical information

indicating a status of each of the one or more file systems;

in response to a predetermined event, obtain critical information associated with the one or more file systems stored in the non-volatile memory immediately before the predetermined event,

determine if one of the one or more file systems has crashed based on the critical information; and

determine if one of the one or more file systems is mounted improperly if it is determined that the one of the one or more file systems has crashed.

11. The non-transitory computer readable storage medium according to claim 10, wherein the critical information associated with each of the one or more file systems comprises a data structure indicating the status of a corresponding one of the one or more file systems.

12. The non-transitory computer readable storage medium according to claim 10, wherein the set of instructions is further operable to initiate a failure recovery process when it is determined that one of the one or more file systems is mounted improperly.

13. The non-transitory computer readable storage medium according to claim 12, wherein the critical information indicates at least one of a normal-off status, a running status and an abnormal-off status, and

wherein the critical information is configured to change from the normal-off status to the running status during a start-up procedure, and

wherein the critical information is configured to change to the normal-off status after a shutting-down procedure, and

wherein the critical information is configured to change to the abnormal-off status when a failure is detected in the file system.

14. The non-transitory computer readable storage medium according to claim 13, wherein the set of instructions is further operable to determine if the file system has been properly shut down after the shutting-down procedure.

15. The non-transitory computer readable storage medium according to claim 10, wherein the set of instructions is operable to persistently store metadata in an area of the non-volatile memory such that the metadata will not be overwritten, wherein the metadata is used to reboot one of the one or more file systems when it is determined that the one of the one or more file systems is mounted improperly.

16. The non-transitory computer readable storage medium according to claim 10, wherein the predetermined event is one during which one of the one or more file systems has crashed and a failure is detected in the one of the one or more file-systems.

17. A dual mode server capable of a dual state operation and single state operation, the dual mode server comprising:

a non-volatile memory configured to persistently store critical information, the critical information indicating a status of each of the one or more file systems;

a processor configured to~~-~~:

in response to a predetermined event, obtain critical information ~~of~~ associated with the one or more file systems stored in the non-volatile memory immediately before the predetermined event,

determine if at least one of the one or more file systems has crashed based on the critical information~~;~~, and

if it is determined that the at least one of the one or more file systems has crashed and the server is in the dual state operation, switch the server from the dual state operation to the single state operation to allow metadata to be flushed to a block device.

18. The dual mode server according to claim 17 wherein the processor is further configured to switch from the single state operation back to the dual state operation when the metadata has been flushed to the block device.

19. A method for organizing critical data in a non-volatile memory comprising:

obtaining a data structure element to represent a file system object when the file system object is opened, the data structure element comprising attribute information and disk lock location of the file system object;

calculating a data block based on the data structure element;

obtaining a root node associated with the data block; and

determining a middle node and a leaf node in response to the leaf node.

20. The method according to claim 19 wherein the file system object is a file or a directory.

21. The method according to claim 19 wherein the file system object is a unix-style file system object and the data structure element is an inode.